THREE-DIMENSIONAL MODELING
FROM ARBITRARY THREE-DIMENSIONAL CURVES

CROSS-REFERENCES TO RELATED APPLICATIONS

5   This application claims priority of US provisional patent application serial number 60/477,933 filed June 13, 2003.

BACKGROUND OF THE INVENTION

Field of the Invention

This invention relates to the field of three-dimensional imaging systems, and more
10   particularly to a method and apparatus for creating three-dimensional models of objects from sets of arbitrary three-dimensional curves obtained from a target surface.

Description of Prior Art

Three-dimensional digitization and modeling systems are commonly used in many *industries* and their applications are numerous. A few examples of such applications are:
15   inspection and measurement of shape conformity in industrial production systems, digitization of clay models for industrial design and styling applications, reverse *engineering of existing parts, interactive visualization of objects in multimedia* applications, and three-dimensional documentation of artwork and artefacts.

The shape of an object is digitized using a ranging sensor that measures the distance
20   between the sensor and a set of points on the surface. From these measurements, three-dimensional coordinates of points on the target surface are obtained in the sensor reference frame. From a given point of view, the ranging sensor can only acquire distance measurements on the visible portion of the surface. To digitize the whole object, the sensor must therefore be moved to a plurality of viewpoints in order to acquire sets of
25   range measurements that cover the entire surface. A model of the object is built by merging and registering the sets of range measurements in a common reference frame. Creating the three-dimensional model of an object is a multistep process usually comprising: (a) acquiring sets of range measurements from a plurality of viewpoints, (b)

roughly aligning the sets of range measurements in a common reference frame, (c) refining the registration of the sets of range measurements in the common reference frame, (d) merging the range measurements in a unique non-redundant model, and (e) building a model of the surface using geometric primitives such as triangles.

5   Three types of ranging sensors are commonly used to digitize the surface of an object: (a) cloud of points sensors which provide unorganized sets of range measurements, (b) range image sensors which provide sets of measurements organized into regular matrices, and (c) curve sensors which provide range measurements organized into sets of three-dimensional curves, e.g. profiles. Cloud of points and range image sensors offer the
10   advantage of providing a high density of measurements from a single viewpoint. They are however bulky and cumbersome to move from one viewpoint to the next. Due to their greater simplicity and robustness, curve sensors are usually favored for digitizing the entire shape of objects. These sensors can be easily moved from one viewpoint to the next, by either a mechanical system or simply an operator holding the sensor in hand.

15   Numerous examples of ranging sensor devices are described in the art. "Portable digital 3-d imaging system for remote sites" by J.-A. Beraldin et al., published in proceedings of the 1998 IEEE International Symposium on Circuit and Systems, Monterey, CA, USA: 326-333. May 31-June 3, 1998, describes a compact laser stripe range sensor that can be mounted, for instance on a translational or rotational stage. "A Self-Referenced Hand-
20   Held Range Sensor" by P. Hebert, published in Proceedings of the IEEE International Conference on Recent Advances in 3-D Digital Imaging and Modeling, Quebec, pp. 5-12, May 2001, presents a hand-held range sensor gathering a set of curves issued from a crosshair laser projection pattern. In "A Review of 20 Years of Ranges Sensor Development" by F. Blais, published in, Videometrics VII, in Proceedings of SPIE-IS&T
25   Electronic Imaging , SPIE Volume 5013 (2003), pp 62-76, NRC 44965, several range sensors are described, among these a range scanner collecting both 3-D measurements and color (R,G,B) properties is further described in "Digital 3-D Imaging, theory and applications" by M. Rioux, published in Proceedings of Videometrics III, SPIE vol. 2350, pp.2-15, 1994.

30   Using curve sensors with current three-dimensional modeling systems requires that the sensor be moved in an organized and regular motion perpendicular to the direction of the

measured profile. This, to simulate a range image that can be processed in the same way the output of a range image sensor would be. Since the range of motion is restricted to a single axis, the operation must be repeated multiple times to simulate the acquisition of range images from multiple viewpoints. In addition to imposing a regular unidirectional

5   motion of the sensor, the main drawback with current systems that process data in this manner is that the alignment error of each curve cannot be individually corrected. As the set of curves acquired in a given motion sequence is processed as a rigid range image, only the global alignment error of the entire set of curves can be corrected by the modeling system, thus limiting the accuracy of the reconstructed surface.

10 No method has been developed in the art for creating a surface model from arbitrary three-dimensional curves. U.S. Pat. No. 5,946,645 issued Aug. 31, 1999 to Marc Rioux and Patrick Hebert describes a method for individually refining the alignment of arbitrary three-dimensional profiles. The method claimed relies on measuring the distance between each profile and every other profile in the set. Because of this combinatorial complexity,

15 the method cannot currently be used in practice with the high number of curves typically needed to model an entire object. Furthermore, this patent does not describe any means for creating a model of the surface from the three-dimensional curves once their alignment has been refined. The use of this curve registration method on data acquired from a hand-held sensor is described in "Toward a hand-held laser range scanner:

20 integrating observation-based motion compensation" by P. Hebert et al., published in Proceedings of SPIE, volume 3313, pages 2-13, January 1998.

A volumetric method for reconstructing surface models from unorganized clouds of three-dimensional points is described in "Surface Reconstruction from Unorganized Points" by H. Hoppe et al., published in SIGGRAPH '92 Proceedings, volume 26, pages

25 71-78, July 1992. The volumetric representation used by this method only contains the distance to the surface, which is sufficient to reconstruct the surface but does not contain enough information to refine alignment errors of data sets. This method is general and can be used to create a surface model for a set of arbitrary three-dimensional curves by converting it into an unorganized cloud of points. This process is difficult to implement in

30 practice because it requires that the entire target surface be measured with a high and relatively uniform density. Doing so also eliminates the rigid arrangement of each curve in the set. This loss of information negatively impacts the quality and accuracy of the

reconstructed surface because it becomes impossible to correct the alignment of curves. No method for refining the alignment of the three-dimensional data is described. A specialization of this surface reconstruction method for merging range images is described in "A Volumetric Method for Building Complex Models from Range Images" by B. Curless et al., published in SIGGRAPH '96, Proceedings, pages 303-312, 1996. This is a volumetric method that reconstructs the surface incrementally by adding range images to the model. The method exploits the structure of the images and cannot be used with arbitrary three-dimensional curves. As with the more general approach described by Hoppe, no method for refining the alignment of the three-dimensional data is described.

A method for reconstructing a surface from data acquired with a hand-held curve sensor is described in "Geometric Fusion for a Hand-Held 3D Sensor" by A. Hilton et al., published in Machine Vsion and Applications, 12:44-51, 2000. This method requires that the sensor is moved in a regular and organized motion in such a way to provide parallel curves that are merged into range images. Instead of directly using the three-dimensional curves to build the surface, this method must combine the curves into simulated range images. The main drawback of the method is that it poses severe constraints on the range of motion of the sensor. Also, since the curves are merged into images, the alignment error of each curve cannot be individually corrected.

"Registration and Integration of Multiple Range Images by Matching Signed Distance Fields for Object Shape Modeling" by T. Masuda published in Computer Vision and Image Understanding 87, pages 51-65, 2002 describes a volumetric method for surface reconstruction and alignment refinement based on signed distance fields. This method is however specific to range images and cannot be generalized to arbitrary three-dimensional curves because signed distance fields cannot be constructed from single curves.

There therefore exists a need for new methods for creating three-dimensional models of objects from sets of arbitrary three-dimensional entities obtained from a target surface and eliminating the need for moving the sensor in organized and regular motions to simulate the creation of range images.

SUMMARY OF THE INVENTION

It is an object of the present invention to overcome the problems mentioned previously.

In a preferred embodiment, the present invention provides a method for ascertaining the three-dimensional shape of a target surface using a hand-held ranging sensor to obtain a plurality of arbitrary overlapping profiles.

The present invention relates to a method for creating three-dimensional models of objects from sets of arbitrary three-dimensional entities obtained from target surfaces. It also provides an efficient method for individually refining the alignment of curves to improve the accuracy of the surface model with a linear complexity with respect to the number of curves. The principle behind the invention is that a set of three-dimensional entities, at their approximate positions, creates a field from which the surface can be extracted. The field is constructed in a manner such that the three-dimensional entities are attracted toward the extracted surface. This attraction is used to accurately register each three-dimensional entity with respect to extracted surface. Through iterations, both the field and the entity positions are refined.

According to one broad aspect of the present invention, there is provided a method for reconstructing surfaces from a single or a plurality of arbitrary three-dimensional entities obtained from a target surface. The method comprises: obtaining a set of at least one three-dimensional entity, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinate of the point on the target surface; constructing a volumetric implicit representation of the target surface in the form of a vector field, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; placing each three-dimensional entity in the vector field and updating the vector field accordingly; and Reconstructing the target surface from the information contained in the vector field.

Preferably, at least one of the three-dimensional entities is an organized set of three-dimensional points that are part of one or a plurality of three-dimensional curves.

According to another broad aspect of the present invention, there is provided a method for refining the alignment of arbitrary three-dimensional entities obtained from a target surface. The method comprises:

(a) obtaining a set of at least two three-dimensional entities, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinate of the point on the target surface;

(b) constructing a volumetric implicit representation of the target surface in the form of a vector field, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface;

(c) placing at least two three-dimensional entities in the vector field and updating the vector field accordingly;

(d) selecting at least one three-dimensional entity placed in the vector field and obtaining a subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points;

(e) for each control point in each selected three-dimensional entity, computing a contribution to a cost function, this contribution being a function of at least the vector field and the three-dimensional coordinate of the control point;

(f) for each selected three-dimensional entity, computing a new position that optimizes its corresponding cost function; and

(g) placing each selected three-dimensional entity in the vector field at its newly computed position and updating the vector field accordingly.

According to a further aspect of the present invention, the steps (c) and (d) of the previous method can be modified for:

(c) placing at least one three-dimensional entity in the vector field and updating the vector field accordingly;

(d) selecting at least one of the three-dimensional entities not yet placed in the vector field, placing the selected three-dimensional entities in the vector field

without updating the field and obtaining a subset of three-dimensional points on each of the selected entities, three-dimensional points in these subsets being called control points.

According to still another broad aspect of the present invention, there is provided a
5   method for ascertaining the three-dimensional shape of a target surface. The method comprises using a ranging sensor to produce a set of three-dimensional entities, a method according to present invention to reconstruct the target surface, and a method according to the present invention to improve the quality and accuracy of the reconstructed surface.

According to another broad aspect of the present invention, there is provided a system for
10   reconstructing surfaces from a single or a plurality of arbitrary three-dimensional entities obtained from a target surface comprising: a three-dimensional entity provider for obtaining a set of at least one three-dimensional entity, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinates of the point on the target surface; an implicit representation
15   constructor for constructing a volumetric implicit representation of the target surface in the form of a vector field, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; for placing each three-dimensional entity in the vector field; and for updating the vector field accordingly; and a target surface reconstructor for reconstructing the target surface from
20   the information contained in the vector field.

According to a further broad aspect of the present invention, there is provided a system for refining an alignment of arbitrary three-dimensional entities obtained from a target surface, comprising: a three-dimensional entity provider for obtaining a set of at least two three-dimensional entity, each entity being a set of three-dimensional points, each three-
25   dimensional point containing at least the three-dimensional coordinates of the point on the target surface; an implicit representation constructor for constructing a volumetric implicit representation of the target surface in the form of a vector field, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; for placing at least two three-dimensional
30   entity in the vector field; and for updating the vector field accordingly; and a control point selector for selecting at least one three-dimensional entity placed in the vector field

and obtaining a subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points; a cost function calculator for computing, for each control point in each selected three-dimensional entity, a contribution to a cost function, the contribution being a function of

5   at least the vector field and the three-dimensional coordinate of the control point; a new position calculator for computing, for each selected three-dimensional entity, a new position that optimizes its corresponding cost function; wherein the implicit representation constructor places each selected three-dimensional entity in the vector field at its newly computed position and updates the vector field accordingly.

10   According to still a further broad aspect of the present invention, there is provided a system for refining an alignment of arbitrary three-dimensional entities obtained from a target surface, comprising: a three-dimensional entity provider for obtaining a set of at least two three-dimensional entity, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinates of the

15   point on the target surface; an implicit representation constructor for constructing a volumetric implicit representation of the target surface in the form of a vector field, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; for placing at least one three-dimensional entity in the vector field; and for updating the vector field accordingly; and a control

20   point selector for selecting at least one of the three-dimensional entities not yet placed in the vector field, wherein the implicit representation constructor places the selected three-dimensional entities in the vector field without updating the field, and obtaining a subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points; a cost function calculator

25   for computing, for each control point in each selected three-dimensional entity, a contribution to a cost function, the contribution being a function of at least the vector field and the three-dimensional coordinate of the control point; a new position calculator for computing, for each selected three-dimensional entity, a new position that optimizes its corresponding cost function; wherein the implicit representation constructor places each

30   selected three-dimensional entity in the vector field at its newly computed position and updates the vector field accordingly.

In a preferred embodiment, the invention is used to process three dimensional curves obtained from a hand held sensor such as that described in "A Self-Referenced Hand-Held Range Sensor" by P. Hebert, published in Proceedings of the IEEE International Conference on Recent Advances in 3-D Digital Imaging and Modeling, Quebec, pp. 5-12,

5 May 2001.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other features, aspects and advantages of the present invention will become better understood with regard to the following description and accompanying drawings

10 wherein:

Fig. 1 represents a flowchart describing the method for surface reconstruction from arbitrary three-dimensional curves followed by a registration of three-dimensional curves to the reconstructed surface;

Fig. 2 represents the initialization step;

15 Fig. 3 represents the preprocessing step;

Fig. 4 represents the model reconstruction step;

Fig. 5 represents the integration of a single curve into the volumetric structure as a step for model reconstruction;

Fig. 6 represents the determination of the distance d between a voxel and a line segment

20 in the integration of a single curve step;

Fig. 7 represents the curve registration step;

Fig. 8 represents the registration procedure for a single curve;

Fig. 9 represents the extrapolation of the matching point m_j for the control point c_j in the registration procedure for a single curve;

Fig. 10 represents a flowchart where each curve is registered prior to its integration to the volumetric structure;

Fig. 11 represents a flowchart of the main steps of one embodiment of the invention;

Fig. 12 represents a flowchart of the main steps of a second embodiment of the invention;

5   Fig. 13 represents a flowchart of the main steps of a third embodiment of the invention; and

Fig. 14 represents a block diagram of the main components of the present invention.

DESCRIPTION OF PREFERRED EMBODIMENT

10   The present invention now will be described more fully hereinafter. However, this invention may be embodied in many different forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the invention to those skilled in the art.

15   Figure 1 shows a flowchart describing the functioning of the method for surface reconstruction from arbitrary 3D curves as well as for registration of 3D curves to the reconstructed surface.

INITIALISATION

Initialization step **100**, illustrated in Figure 2, requires two user-supplied parameters:
20   bounding box of the object to be reconstructed and the resolution of the volumetric lattice of points (volumetric structure).

The   bounding   box   is   defined   as   coordinates $b_{min} = [x_{min}, y_{min}, z_{min}]^T$ and $b_{max} = [x_{max}, y_{max}, z_{max}]^T$ of two opposite corners of the box that completely contains the object, such that $x_{min} < x_{max}$, $y_{min} < y_{max}$ and $z_{min} < z_{max}$.

Resolution Δ of the volumetric structure is defined as the distance between two neighbouring lattice points.

Volumetric lattice V of points is defined 202 as a set V of points (voxels)

$$V = \left\{ [x_i, y_j, z_k]^T \mid x_i = x_{min} + i\Delta, y_j = y_{min} + j\Delta, z_k = z_{min} + k\Delta \right\},$$

$$i = 0, \ldots, N_x, \quad j = 0, \ldots, N_y, k = 0, \ldots, N_z, \text{ where}$$

$$N_x = \left\lceil \frac{x_{max} - x_{min}}{\Delta} \right\rceil, \quad N_y = \left\lceil \frac{y_{max} - y_{min}}{\Delta} \right\rceil \text{ and } N_z = \left\lceil \frac{z_{max} - z_{min}}{\Delta} \right\rceil.$$

Operator $\lceil x \rceil$ represents smallest integer greater than x.

A 3×3 matrix K (covariance matrix) is attached to each lattice point and all elements of the matrix K are initialized to zero 203. Also, to each lattice point, a vector g (distance vector) is attached as well as a weighting factor ω both initialized to zero. This volumetric structure represents implicitly the reconstructed surface.

INPUT DATA

Input 101 to the reconstruction and registration algorithm is a set $C = \{C_1, C_2, \ldots, C_N\}$ of 3D curves measured on the surface of an object. A curve C_l is defined as a set of 3D points $C_l = \{p_1, p_2, \ldots, p_{M_l}\}$, $p_k = [x_k \quad y_k \quad z_k]^T$ such that the real surface curve can be approximated as a set of line segments $\{p_1 p_2, p_2 p_3, \ldots, p_{M_l-1} p_{M_l}\}$.

PREPROCESSING

The only preprocessing step 102, Figure 3, is the estimation of tangents at each point of all curves 304. At a point p_k of curve $C_l = \{p_1, p_2, \ldots, p_{M_l}\}$ the tangent t_k can be estimated as:

$$t_k = \frac{1}{2} \left[\frac{p_k - p_{k-1}}{\|p_k - p_{k-1}\|} + \frac{p_{k+1} - p_k}{\|p_{k+1} - p_k\|} \right]$$

Optionally, the tangents can be filtered by averaging them with tangents at neighbouring points:

$$t_k \leftarrow \frac{1}{3}[t_{k-1} + t_k + t_{k+1}]$$

After the preprocessing step, a set of tangents $T_i = \{t_1, t_2, \ldots, t_{M_i}\}$ is assigned to each curve C_i 307. To distinguish the tangents belonging to different curves, the box 304 specifies t_{ik} as a single tangent, where i is the curve index.

5   MODEL RECONSTRUCTION

Model reconstruction 103, Figure 4, consists of the integration of each curve independently into the volumetric structure 404. The order in which the curves are integrated is not important. The final result is the same regardless the order of the curves.

INTEGRATION OF A SINGLE CURVE INTO THE VOLUMETRIC STRUCTURE

10   The integration of a single curve into the volumetric structure 404 is depicted as a flowchart diagram in Figure 5. For the purpose of integration of curves into the volumetric structure, a curve $C_i = \{p_1, p_2, \ldots, p_{M_i}\}$ is considered as a set of line segments $\{p_1 p_2, p_2 p_3, \ldots, p_{M_i-1} p_{M_i}\}$. A curve is always integrated one line segment at the time 504. In the following, the computation is explained for a single line segment $p_k p_{k+1}$. The same

15   procedure is repeated for all other line-segments contained in the curve 514.

The integration procedure consists in updating the covariance matrices and the distance vectors in the neighbourhood of the line segment. The size of the neighbourhood ε is a user-specified parameter and cannot be smaller then the $\sqrt{3}\Delta$, where Δ is the resolution of the volumetric structure, as defined above.

20   The first step of integration 404 is to compute the bounding box B for the line segment. The bounding box is defined as coordinates $b_{min} = [x_{min}, y_{min}, z_{min}]^T$ and $b_{max} = [x_{max}, y_{max}, z_{max}]^T$ of two opposite corners of a rectangular box such that

$$x_{min} < x_{max},$$
$$y_{min} < y_{max} \text{ and}$$
25   $$z_{min} < z_{max},$$

where

$$x_{\min} = \min(x_k - \varepsilon, x_{k+1} - \varepsilon), \quad x_{\max} = \max(x_k + \varepsilon, x_{k+1} + \varepsilon)$$

$$y_{\min} = \min(y_k - \varepsilon, y_{k+1} - \varepsilon), \quad y_{\max} = \max(y_k + \varepsilon, y_{k+1} + \varepsilon)$$

$$z_{\min} = \min(z_k - \varepsilon, z_{k+1} - \varepsilon), \quad z_{\max} = \max(z_k + \varepsilon, z_{k+1} + \varepsilon)$$

5   Let L be the set of all lattice points (voxels) within the bounding box B $L = \{v_j \in V \mid v_j \in B, j = 1, \ldots, Q\}$, where V is the set of lattice points 202. For each voxel $v_j \in L$ it is first verified that the voxel v_j is in-between two planes Π_1 and Π_2 defined as planes that contain points $\mathbf{p}_k$ and $\mathbf{p}_{k+1}$ and whose normals are equal to tangents $\mathbf{t}_k$ and $\mathbf{t}_{k+1}$ respectively 510. If not, the next voxel is selected 507.

10   To compute the distance, a line segment $\mathbf{p}_k \mathbf{p}_{k+1}$ is parameterized as:

$$l(u) = \mathbf{p}_k + u(\mathbf{p}_{k+1} - \mathbf{p}_k), 0 \le u \le 1.$$

The tangents at the two end points are then interpolated as:

$$t(u) = \mathbf{t}_k + u(\mathbf{t}_{k+1} - \mathbf{t}_k), 0 \le u \le 1.$$

The normal $\mathbf{n}(u)$ at each point of a line segment is defined as the vector being contained

15   in the plane $\mathbf{p}_k \mathbf{p}_{k+1} v_j$ and perpendicular to the tangent $t(u)$. The normal $\mathbf{n}(u)$ and is also interpolated over the line segment in the same manner:

$$\mathbf{n}(u) = \mathbf{n}_k + u(\mathbf{n}_{k+1} - \mathbf{n}_k), 0 \le u \le 1.$$

Normals $\mathbf{n}_k$ and $\mathbf{n}_{k+1}$ are unit vectors perpendicular to tangents $\mathbf{t}_k$ and $\mathbf{t}_{k+1}$ respectively and contained in plane $\mathbf{p}_k \mathbf{p}_{k+1} v_j$.

20   The distance d between voxel v_j and the line segment $\mathbf{p}_k \mathbf{p}_{k+1}$ is defined as the distance between v_j and the point $\mathbf{p}_c$ whose normal $\mathbf{n}_c$ passes through v_j. This is illustrated in Figure 6. The distance is computed as the distance for which the area of the triangle $\mathbf{q}_k \mathbf{q}_{k+1} v_j$ is zero, i.e. that the cross-product of $\mathbf{q}_k - v_j$ and $\mathbf{q}_{k+1} - v_j$ is zero, where

$$\mathbf{q}_k = d\mathbf{n}_k + \mathbf{p}_k,$$

25

$$\mathbf{q}_{k+1} = d\mathbf{n}_{k+1} + \mathbf{p}_{k+1}.$$

The distance d is obtained from the equation

$$\mathbf{a} + d\mathbf{b} + d^2\mathbf{c} = 0 \,,$$

where

$$\mathbf{a} = (\mathbf{p}_k - \mathbf{v}_j) \times (\mathbf{p}_{k+1} - \mathbf{v}_j) \,,$$

$$\mathbf{b} = \mathbf{n}_{k+1} \times (\mathbf{p}_k - \mathbf{v}_j) + \mathbf{n}_k \times (\mathbf{p}_{k+1} - \mathbf{v}_j) \,,$$

$$\mathbf{a} + d\mathbf{b} + d^2\mathbf{c} = 0 \,.$$

Preceding equation is a system of three quadric equations with a single unknown d. Any of these equations can be used to compute d after making sure that the chosen parameters $\mathbf{a}, \mathbf{b}, \mathbf{c}$ do not vanish altogether. The chosen equation can have up to two real roots. If the number of roots is zero, the voxel $\mathbf{v}_j$ is rejected and next voxel is selected 507. If there are two real roots, the root of interest is the one with the smallest positive value.

Using the distance computed above, parameter u_c is computed as:

$$u_c = \frac{\|\mathbf{q}_k - \mathbf{v}_j\|}{\|\mathbf{q}_{k+1} - \mathbf{q}_k\|} \,.$$

Next, it is verified whether the distance d is smaller then the maximal allowed distance ε. If not, next voxel is selected 507.

Finally, the covariance matrix K_j at voxel $\mathbf{v}_j$ is updated as follows:

$$K_j \leftarrow K_j + \mathbf{t}(u_c)\mathbf{t}(u_c)^T$$

Vector $\mathbf{g}_j$ is updated as:

$$\mathbf{g}_j \leftarrow \mathbf{g}_j + e^{-\frac{d^2}{0.5\Delta}}(\mathbf{v}_j - \mathbf{p}_c)$$

Weighting value ω_j is updated as:

$$\omega_j \leftarrow \omega_j + e^{-\frac{d^2}{0.5\Delta}}$$

The whole procedure is repeated for the remaining voxels 514 as well as for the remaining line segments 515.

The resulting volumetric structure contains an implicit representation of the reconstructed surface (reconstructed model). The covariance matrices K, and distance vectors $\mathbf{g}$, represent implicitly a vector field whose value is the direction and the distance toward the closest point on the reconstructed model.

5 CURVE REGISTRATION

Each curve is registered independently 604 to the reconstructed model that is implicitly represented in the volumetric structure. Flowchart in Figure 7 illustrates the overall procedure.

REGISTRATION OF A SINGLE CURVE

10 Figure 8 shows registration procedure for a single curve $C_i = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_{M_i}\}$. First a subset W_i of points in C_k is chosen 702 as a set of control points

$$W_i \subseteq C_i = \{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_{Q_i}\}$$

(The set W_i can be equal to C_i)

15 The following procedure is repeated for all control points $\mathbf{c}_j$. First the closest voxel $\mathbf{v}_c$ to the control point $\mathbf{c}_j$ is found. Then the direction $\mathbf{f}$ from point $\mathbf{c}_j$ toward the reconstructed model is computed using the covariance matrix K_c at voxel $\mathbf{v}_c$. The direction $\mathbf{f}$ is the eigenvector associated with the smallest eigenvalue of matrix K_c / ω_c, where ω_c is the weighting factor at voxel $\mathbf{v}_c$. The distance e to the reconstructed model

20 is obtained as $\langle \mathbf{f}, \mathbf{g}_c \rangle / \omega_c$. The matching point $\mathbf{m}_j$ for the control point $\mathbf{c}_j$ is extrapolated as

$$\mathbf{m}_j = \mathbf{c}_j + \mathbf{f}[e + < \mathbf{f}, (\mathbf{v}_c - \mathbf{c}_j) >]$$

which is illustrated in figure 9. The value $e\mathbf{f}$ is a vector that indicates the direction and the distance toward the closest point on the reconstructed model from the voxel $\mathbf{v}_c$.

25 Once the matching point is obtained for all control points, a rigid transformation Φ that best aligns two sets of points $\{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_J\}$ and $\{\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_J\}$ is computed. This is

accomplished using standard techniques, for example quaternions. The transformation matrix Φ is then applied to all curve points:

$$\mathbf{p}_l \leftarrow \Phi \mathbf{p}_l, \quad l = 0,...,M_i.$$

An example of a cost function Ψ that can be used to compute the rigid transformation Φ is a sum of squared distances between corresponding points, i.e.

$$\Psi = \sum_{k=1}^{J} \|\mathbf{c}_i - \mathbf{m}_i\|^2$$

The cost function can be modified to exploit surface proprieties to improve the quality of registration. For example, the following cost function utilises color $\Gamma = \begin{bmatrix} R & G & B \end{bmatrix}^T$ associated with each point:

$$\Psi' = \sum_{k=1}^{J} \lambda_k \|\mathbf{c}_k - \mathbf{m}_k\|^2,$$

where λ_k is a weighting factor dependent on similarity of colors:

$$\lambda_k = \begin{cases} 1 & \text{if colors } \Gamma_{ck} \text{ and } \Gamma_{mk} \text{ are similar} \\ 0 & \text{otherwise} \end{cases}.$$

Γ_{ck} and Γ_{mk} are colors associated with points $\mathbf{c}_k$ and $\mathbf{m}_k$ respectively.

Prior to computing matching points it has to be verified whether the eigenvalues $\lambda_1 < \lambda_2 < \lambda_3$ of matrix K_c / ω_c satisfy the following relation $\lambda_2 > 0.05$ and $\lambda_1 < 0.5\lambda_2$. These values are provided as indications. They mean that the local distribution of tangents is nearly planar. If not, both control point $\mathbf{c}_j$ and matching point $\mathbf{m}_j$ are rejected. When more than a given number of points are rejected (for example 70%) then the whole curve is rejected.

If the average displacement of curve points after the transformation has been applied is larger then a threshold (for example 1% of the grid resolution Δ) the whole procedure is repeated 710.

SURFACE EXTRACTION

The surface extraction proceeds by converting the implicit surface representation in the volumetric structure into scalar distance field. This is performed by computing the

direction $\mathbf{f}_j$ and distance e from each voxel $\mathbf{v}_j$ toward the reconstructed model if the $\omega > 0$ at voxel $\mathbf{v}_j$. The direction $\mathbf{f}$ is eigenvector associated with the smallest eigenvalue of matrix K_j / ω_j, where K_j is covariance matrix and ω is the weighting factor at voxel $\mathbf{v}_j$. The distance e to the reconstructed model is obtained as $\langle \mathbf{f}_j, \mathbf{g}_j \rangle / \omega_j$.

5   Distance e at each voxel represents a scalar field. A Marching Cubes algorithm, for instance, is applied to this scalar field in order to obtain a triangulated surface (represented as a set of triangles).

ALTERNATIVE EMBODIMENTS AND APPLICATIONS

10   Reconstruction of additional surface properties

Presented method can be used to reconstruct not only model geometry, but also different surface properties including but not limited to color, intensity, texture and temperature. To allow reconstruction of additional object properties the following modifications are

15   required:

Each curve is assumed to have a surface property assigned to each curve point, i.e. for a curve $C_i = \{ \mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_{M_i} \}$ a property $\mathbf{h}_i$ is assigned to each point $\mathbf{p}_i$.

During the initialization step **100** a single or multiple zero valued properties **H** are attached to each lattice point. Surface properties can be either scalar or vector valued.

20   During the integration of curves **404**, property value $\mathbf{H}_j$ at voxel $\mathbf{v}_j$ is updated as:

$$\mathbf{H}_j \leftarrow \mathbf{H}_j + \omega_j (\mathbf{h}_i + u_c (\mathbf{h}_{i+1} - \mathbf{h}_i)).$$

During the extraction of the triangulated surface (by Marching Cubes), property value interpolated between two voxels closest to each triangle vertex.

Alternative configuration: Incremental registration

25   The two basic building blocks of the method, curve integration **404** and curve registration **604** can be rearranged into a different configuration to accomplish incremental registration where each curve is registered prior to its integration. The flowchart in figure

10 illustrates the principle of functioning. All building blocks remain the same as for the configuration in Figure 1.

Modeling from different surface entities

The method presented above accepts as input a set of non-parallel 3D curves which generates organized set of three-dimensional points. However, the same method can be used to build and update the volumetric structure from various arbitrary three-dimensional entities such as partial surfaces (range images), parallel curves, unorganized clouds of points or any combination of these. Moreover, the three-dimensional points can be measured in a single plane and the coordinate system can be reduced to a two-dimensional system. Required modification for each one of these cases is discussed below.

Modeling from Partial Surfaces (Range images)

Range images are matrices of points measured on the surface of an object in such a way as to allow an approximation of the real surface by a set of connected triangles. Such a surface S is defined as a set of O triangles $S = \{\Lambda_1, \Lambda_2, ..., \Lambda_O\}$, where each triangle Λ_i is defined by three vertices $\mathbf{p}_{i1}$, $\mathbf{p}_{i2}$ and $\mathbf{p}_{i3}$. It is also assumed that a normal $\mathbf{n}_{ij}$ is estimated at each vertex, which can be easily accomplished using standard techniques. The surface is integrated into the volumetric structure one triangle at the time. As for the line segments, a bounding box is estimated for the triangle and for each voxel $\mathbf{v}_j$ within bounding box the closest point $\mathbf{p}_c$ on the triangle is found. The elements attached at the voxel are then updated as follows:

$$K_j \leftarrow K_j + \mathbf{t}_1 \mathbf{t}_2^T$$

Vector $\mathbf{g}_j$ is updated as:

$$\mathbf{g}_j \leftarrow \mathbf{g}_j + e^{-\frac{d^2}{0.5\Delta}}(\mathbf{v}_j - \mathbf{p}_c)$$

Weighting value ω_j is updated as.

$$\omega_j \leftarrow \omega_j + e^{-\frac{d^2}{0.5\Delta}}$$

where $d = \|\mathbf{v}_j - \mathbf{p}_c\|$, and $\mathbf{t}_1$ and $\mathbf{t}_2$ are two vectors, perpendicular to each other as well as to the normal at $\mathbf{p}_c$.

Surface is registered to the reconstructed model by choosing a subset of (or all) vertices of the input surface.

Modeling from parallel curves

In order to allow the reconstruction from parallel curves, the elements attached at each voxel $\mathbf{v}_j$ are updated as follows:

$$K_j \leftarrow K_j + \mathbf{t}_1 \mathbf{t}_2^T + (\mathbf{v}_j - \mathbf{p}_c)(\mathbf{v}_j - \mathbf{p}_c)^T$$

Vector $\mathbf{g}_j$ is updated as:

$$\mathbf{g}_j \leftarrow \mathbf{g}_j + e^{-\frac{d^2}{0.5\Delta}}(\mathbf{v}_j - \mathbf{p}_c)$$

Weighting value ω_j is updated as:

$$\omega_j \leftarrow \omega_j + e^{-\frac{d^2}{0.5\Delta}}.$$

Prior to the registration, or surface extraction the covariance matrix K_c is changed to

$$K_c \leftarrow K_c - \mathbf{g}_j \mathbf{g}_j^T$$

Modeling for unorganized sets of points

An unorganized P set of points is simply a set of 3D Points $P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_{M_i}\}$. Model reconstruction is performed one point at a time. For a point $\mathbf{p}_i$, a set of voxels contained in the sphere centered at $\mathbf{p}_i$ is found first. Radius of sphere is ε. Then the elements attached at each voxel $\mathbf{v}_j$ are updated as follows:

$$K_j \leftarrow K_j + (\mathbf{v}_j - \mathbf{p}_i)(\mathbf{v}_j - \mathbf{p}_i)^T$$

Vector $\mathbf{g}_j$ is updated as:

$$\mathbf{g}_j \leftarrow \mathbf{g}_j + e^{-\frac{d^2}{0.5\Delta}}(\mathbf{v}_j - \mathbf{p}_i)$$

Weighting value ω_j is updated as.

$$\omega_j \leftarrow \omega_j + e^{-\frac{d^2}{0.5\Delta}}$$

where $d = \|\mathbf{v}_j - \mathbf{p}_i\|$.

As shown in Fig. 11, one embodiment of the preferred invention, namely a method for reconstructing surfaces from a single or a plurality of arbitrary three-dimensional entities obtained from a target surface, comprises the following steps: obtaining a set of at least one three-dimensional entity 1100, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinates of the point on the target surface; constructing a volumetric implicit representation of the target surface in the form of a vector field 1101, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; placing each three-dimensional entity in the vector field and updating the vector field accordingly 1102; and reconstructing the target surface from the information contained in the vector field 1103.

In a method for ascertaining the three-dimensional shape of a target surface, an additional step of using a ranging sensor to produce a set of three-dimensional entities 1104 is carried out.

As shown in Fig. 12, another embodiment of the preferred invention, namely a method for refining the alignment of arbitrary three-dimensional entities obtained from a target surface, comprises the following steps : obtaining a set of at least two three-dimensional entities 1200, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinate of the point on the target surface; constructing a volumetric implicit representation of the target surface in the form of a vector field 1201, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; placing at least two three-dimensional entities in the vector field and updating the vector field

accordingly 1202; selecting at least one three-dimensional entity placed in the vector field and obtaining a subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points 1203; for each control point in each selected three-dimensional entity, computing a

5   contribution to a cost function 1204, the contribution being a function of at least the vector field and the three-dimensional coordinate of the control point; for each selected three-dimensional entity, computing a new position 1205 that optimizes its corresponding cost function; and placing each selected three-dimensional entity in the vector field at its newly computed position and updating the vector field accordingly 1206.

10   As shown in Fig. 13, another embodiment of the preferred invention, namely a method for refining the alignment of arbitrary three-dimensional entities obtained from a target surface, comprises the following steps: obtaining a set of at least two three-dimensional entities 1300, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinate of the point on the target

15   surface; constructing a volumetric implicit representation of the target surface in the form of a vector field 1301, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; placing at least one three-dimensional entity in the vector field and updating the vector field accordingly 1302; selecting at least one of the three-dimensional entities not yet placed in the vector

20   field 1303, placing the selected three-dimensional entities in the vector field without updating the field 1304 and obtaining a subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points 1305; for each control point in each selected three-dimensional entity, computing a contribution to a cost function 1306, the contribution being a function of at

25   least the vector field and the three-dimensional coordinate of the control point; for each selected three-dimensional entity, computing a new position 1307 that optimizes its corresponding cost function; and placing each selected three-dimensional entity in the vector field at its newly computed position and updating the vector field accordingly 1308.

30   As shown in Fig. 14, a preferred embodiment of the system for reconstructing surfaces from a single or a plurality of arbitrary three-dimensional entities obtained from a target surface comprises the following main components: a three-dimensional entity provider

1401 for obtaining a set of at least one three-dimensional entity, each entity being a set of three-dimensional points, each three-dimensional point containing at least the three-dimensional coordinates of the point on the target surface; an implicit representation constructor 1402 for constructing a volumetric implicit representation of the target

5   surface in the form of a vector field, each vector in the vector field containing at least the distance to the reconstructed surface and the direction toward the reconstructed surface; for placing each three-dimensional entity in the vector field and for updating the vector field accordingly; and a target surface reconstructor 1403 for reconstructing the target surface from the information contained in the vector field. Preferably, if the system is

10   used to ascertain the three-dimensional shape of a target surface, a ranging sensor 1400 is used to produce the set of three-dimensional entities.

If the system is used to refine the alignment of arbitrary three-dimensional entities obtained from a target surface, the following components are provided: a control point selector 1404 for selecting at least one three-dimensional entity placed in the vector field

15   and obtaining a subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points; a cost function calculator 1405 for computing, for each control point in each selected three-dimensional entity, a contribution to a cost function, the contribution being a function of at least the vector field and the three-dimensional coordinate of the control

20   point; for each selected three-dimensional entity, a new position calculator 1406 for computing a new position that optimizes its corresponding cost function. The implicit representation constructor 1402 is then used to place each selected three-dimensional entity in the vector field at its newly computed position and updating the vector field accordingly.

25   If the system is used to refine the alignment of arbitrary three-dimensional entities obtained from a target surface, , the following components are provided: a control point selector 1404 for selecting at least one of the three-dimensional entities not yet placed in the vector field, using the implicit representation constructor 1402 to place the selected three-dimensional entities in the vector field without updating the field and obtaining a

30   subset of three-dimensional points on each of the selected three-dimensional entities, three-dimensional points in these subsets being called control points; a cost function calculator 1405 for computing, for each control point in each selected three-dimensional

entity, a contribution to a cost function, the contribution being a function of at least the vector field and the three-dimensional coordinate of the control point; a new position calculator 1406 for computing, for each selected three-dimensional entity, a new position that optimizes its corresponding cost function; and using the implicit representation

5 constructor 1402 to place each selected three-dimensional entity in the vector field at its newly computed position and updating the vector field accordingly.

While illustrated in the block diagrams as groups of discrete components communicating with each other via distinct data signal connections, it will be understood by those skilled in the art that the preferred embodiments are provided by a combination of hardware and

10 software components, with some components being implemented by a given function or operation of a hardware or software system, and many of the data paths illustrated being implemented by data communication within a computer application or operating system. The structure illustrated is thus provided for efficiency of teaching the present preferred embodiment.

15 It will be understood that numerous modifications thereto will appear to those skilled in the art. Accordingly, the above description and accompanying drawings should be taken as illustrative of the invention and not in a limiting sense. It will further be understood that it is intended to cover any variations, uses, or adaptations of the invention following, in general, the principles of the invention and including such departures from the present

20 disclosure as come within known or customary practice within the art to which the invention pertains and as may be applied to the essential features herein before set forth, and as follows in the scope of the appended claims.